

# Locally exact modifications of numerical schemes

Jan L. Cieřliński\*

Uniwersytet w Białymstoku, Wydział Fizyki  
ul. Lipowa 41, 15-424 Białystok, Poland

## Abstract

We present a new class of exponential integrators for ordinary differential equations: locally exact modifications of known numerical schemes. Local exactness means that they preserve the linearization of the original system at every point. In particular, locally exact integrators preserve all fixed points and are A-stable. We apply this approach to popular schemes including Euler schemes, implicit midpoint rule and trapezoidal rule. We found locally exact modifications of discrete gradient schemes (for symmetric discrete gradients and coordinate increment discrete gradients) preserving their main geometric property: exact conservation of the energy integral (for arbitrary multidimensional Hamiltonian systems in canonical coordinates). Numerical experiments for a 2-dimensional anharmonic oscillator show that locally exact schemes have very good accuracy in the neighbourhood of stable equilibrium, much higher than suggested by the order of new schemes (locally exact modification sometimes increases the order but in many cases leaves it unchanged).

*PACS Numbers:* 45.10.-b; 02.60.Cb; 02.70.-c; 02.70.Bf

*MSC 2000:* 65P10; 65L12; 34K28

*Key words and phrases:* geometric numerical integration, exact discretization, locally exact methods, linearization-preserving integrators, exponential integrators, discrete gradient method, Hamiltonian systems, linear stability

---

\*e-mail: janek@alpha.uwb.edu.pl

# 1 Introduction

The motivation for introducing “locally exact” discretizations is quite natural. Considering a numerical scheme for a dynamical system with periodic solutions (for instance: a nonlinear pendulum) we may ask whether the numerical scheme recovers the period of small oscillations. For a fixed finite time step the answer is usually negative. However, many numerical scheme (perhaps all of them) admit modifications which preserve the period of small oscillations for a fixed (not necessarily small) time step  $h$ . An unusual feature of our approach is that instead of taking the limit  $h \rightarrow 0$ , we consider the limit  $x_n \approx \bar{x}$  (where  $\bar{x}$  is the stable equilibrium). As the next step we consider a linearization around any fixed  $\bar{x}$  (then, in order to preserve the condition  $x_n \approx \bar{x}$ , some evolution of  $\bar{x}$  is necessary). In other words, we combine two known procedures: the approximation of nonlinear systems by linear equations and explicit exact discretizations of linear equations with constant coefficients. An essential novelty consists in applying these procedures to a modified numerical scheme containing free functional parameters. The case of small oscillations was presented in [8]. Our method works perfectly for discrete gradient schemes [20, 22, 29]. We succeeded in modifying discrete gradient schemes in a locally exact way without spoiling their main geometric property: the exact conservation of the energy integral. The preservation of geometric properties by numerical algorithms is of considerable advantage [14, 17]. Promising results on one-dimensional Hamiltonian systems can be found in [9, 10] (by one-dimensional Hamiltonian system we mean a Hamiltonian system with one degree of freedom). In this paper we extend our approach on the case of multidimensional canonical Hamiltonian systems. Moreover, we present locally exact modifications of forward and backward Euler schemes, implicit midpoint rule and trapezoidal rule.

A notion closely related to our local exactness has been proposed a long time ago [27], see also [21]. Recently, similar concept appeared under the name of linearization-preserving preprocessing [23], see also below (section 2.2). Our approach has also some similarities with the Mickens approach [24], Gautschi-type methods [16] and, most of all, with the exponential integrators technique [3, 15, 25]. The definition of an exponential integrator is so wide (e.g., “a numerical method which involves an exponential function of the Jacobian”, [15]) that our schemes can be considered as special exponential integrators. In spite of some similarities and overlappings, our approach differs from all methods mentioned above. In particular, according to our best

knowledge, discrete gradient schemes have never been treated or modified in the framework of exponential integrators and/or linearization-preserving preprocessing.

## 2 Locally exact numerical schemes

We consider an ordinary differential equation (ODE) with the general solution  $\mathbf{x}(t)$  (satisfying the initial condition  $\mathbf{x}(t_0) = \mathbf{x}_0$ ), and a difference equation with the general solution  $\mathbf{x}_n$ . The difference equation is the exact discretization of the considered ODE if  $\mathbf{x}_n = \mathbf{x}(t_n)$ .

### 2.1 Exact discretization of linear systems

It is well known that any linear ODE with constant coefficients admits the exact discretization in an explicit form [28], see also [1, 7, 24]. We summarize these results as follows, compare [10] (Theorem 3.1).

**Proposition 2.1.** *Any linear equation with constant coefficients, represented in the matrix form by*

$$\frac{d\mathbf{x}}{dt} = A\mathbf{x} + \mathbf{b} , \quad (2.1)$$

(where  $\mathbf{x} = \mathbf{x}(t) \in \mathbb{R}^d$ ,  $\mathbf{b} = \text{const} \in \mathbb{R}^d$  and  $A$  is a constant invertible real  $d \times d$  matrix) admits the exact discretization given by

$$\mathbf{x}_{n+1} - \mathbf{x}_n = (e^{h_n A} - I)A^{-1}(A\mathbf{x}_n + \mathbf{b}) , \quad (2.2)$$

where  $h_n = t_{n+1} - t_n$  is the time step and  $I$  is the identity matrix.

**Corollary 2.2.** *We may look at the exact discretization (2.3) as a modification of the forward Euler scheme. Indeed, we can rewrite (2.2) as*

$$\Delta_n^{-1}(\mathbf{x}_{n+1} - \mathbf{x}_n) = A\mathbf{x}_n + \mathbf{b} , \quad (2.3)$$

where  $\Delta_n$ , defined by

$$\Delta_n = A^{-1}(e^{h_n A} - I) , \quad (2.4)$$

is a matrix “perturbation” of the time step  $h_n$ . Indeed,  $\Delta_n = h_n I + O(h_n^2)$ .

Exact discretizations found direct application in numerical treatment of the classical Kepler problem [4, 5, 6]. The exact discretization of the harmonic oscillator equation can also be used to in the integration of some partial differential equations by Fourier transformation [6].

The central topic of our paper is another fruitful direction of using exact integrators, namely the so called locally exact discretizations [6, 9].

## 2.2 Local exactness

Motivated by the results of [8, 9] we propose the following definition.

**Definition 2.3.** *A numerical scheme  $\mathbf{x}_{n+1} = \Phi(\mathbf{x}_n, h_n)$  for an autonomous equation  $\dot{\mathbf{x}} = F(\mathbf{x})$  is locally exact at  $\bar{\mathbf{x}}$  if its linearization around  $\bar{\mathbf{x}}$  is identical with the exact discretization of the differential equation linearized around  $\bar{\mathbf{x}}$ .*

The simplest choice is  $\bar{x} = x_0$ , where  $V'(x_0) = 0$  (small oscillations around the stable equilibrium). In this case  $\delta_n$  does not depend on  $n$ . The resulting scheme, known as MOD-GR (compare [10, 11]), was first presented in [8]. In [9] we considered the case  $\bar{x} = x_n$  (GR-LEX) and its symmetric (time-reversible) modification  $\bar{x} = \frac{1}{2}(x_n + x_{n+1})$  (GR-SLEX). Note that in both cases  $\bar{x}$  changes at every step. The scheme MOD-GR is locally exact at a stable equilibrium only, GR-LEX is locally exact at  $x_n$  (for any  $n$ ), and, finally, GR-SLEX is locally exact at  $\frac{1}{2}(x_n + x_{n+1})$  (for any  $n$ ). In the case of implicit numerical schemes the function  $\Phi(\mathbf{x}_n, h_n)$  is, of course, an implicit function.

**Definition 2.4.** *A numerical scheme  $\mathbf{x}_{n+1} = \Phi(\mathbf{x}_n, h_n)$  for an autonomous equation  $\dot{\mathbf{x}} = F(\mathbf{x})$  is locally exact if there exists a sequence  $\bar{\mathbf{x}}_n$  such that  $\bar{\mathbf{x}}_n - \mathbf{x}_n = O(h_n)$  and the scheme is locally exact at  $\bar{\mathbf{x}}_n$  (for any  $n$ ).*

Therefore GR-LEX and GR-SLEX are locally exact. Similar concept (“linearization-preserving” schemes) has been recently formulated in [23] (Definition 2.1). An integrator is said to be linearization-preserving if it is linearization preserving at all fixed points. This definition is weaker than our Definition 2.4. All locally exact schemes are linearization-preserving but, in general, linearization-preserving schemes do not have to be locally exact in our sense. For instance, the scheme MOD-GR (see [8, 10]) is linearization-preserving (provided that  $V(x)$  has only one stable equilibrium) but is not locally exact. The problem of finding the best sequence  $\bar{\mathbf{x}}_n$  for a given numerical scheme seems to be interesting but has not been considered yet.

## 2.3 Exact discretization of linearized equations

As an immediate consequence of Definition 2.4 we have that the exact discretization of the linearization of a given nonlinear system is locally exact. We confine ourselves to autonomous systems of the form

$$\dot{\mathbf{x}} = F(\mathbf{x}) \quad (2.5)$$

where  $\mathbf{x}(t) \in \mathbb{R}^d$ . If  $\mathbf{x}$  is near a fixed  $\bar{\mathbf{x}}$ , then (2.5) can be approximated by

$$\dot{\boldsymbol{\xi}} = F'(\bar{\mathbf{x}})\boldsymbol{\xi} + F(\bar{\mathbf{x}}) \quad (2.6)$$

where  $F'$  is the Fréchet derivative (Jacobi matrix) of  $F$  and

$$\boldsymbol{\xi} = \mathbf{x} - \bar{\mathbf{x}} . \quad (2.7)$$

The exact discretization of the approximated equation (2.6) is given by

$$\boldsymbol{\xi}_{n+1} = e^{h_n F'(\bar{\mathbf{x}})} \boldsymbol{\xi}_n + \left( e^{h_n F'(\bar{\mathbf{x}})} - I \right) (F'(\bar{\mathbf{x}}))^{-1} F(\bar{\mathbf{x}}) , \quad (2.8)$$

provided that  $\det F'(\bar{\mathbf{x}}) \neq 0$ . This assumption, however, is not necessary because function

$$\varphi_1(x) := \frac{e^x - 1}{x} \quad (\text{and } \varphi_1(0) := 1) \quad (2.9)$$

is analytic also at  $x = 0$ .

**Proposition 2.5.** *The exact discretization of the linearization of  $\dot{\mathbf{x}} = F(\mathbf{x})$  around  $\bar{\mathbf{x}}$  is given by:*

$$\mathbf{x}_{n+1} - \bar{\mathbf{x}} = e^{h_n F'(\bar{\mathbf{x}})} (\mathbf{x}_n - \bar{\mathbf{x}}) + h_n \varphi_1(h_n F'(\bar{\mathbf{x}})) F(\bar{\mathbf{x}}) . \quad (2.10)$$

*The scheme (2.10) is locally exact.*

If we put  $\bar{\mathbf{x}} = \mathbf{x}_n$  into (2.10) (thus changing  $\bar{\mathbf{x}}$  at each step), then  $\boldsymbol{\xi}_n = 0$  and we obtain:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \left( e^{h_n F'(\mathbf{x}_n)} - I \right) (F'(\mathbf{x}_n))^{-1} F(\mathbf{x}_n) \quad (2.11)$$

This method is well known as the exponential difference equation (see [27], formula (3.4)) or, more recently, as the exponential Euler method [25].

### 3 Locally exact modifications of popular one-step numerical schemes

In order to illustrate the concept of locally exact modifications, we apply this procedure to the following class of numerical methods:

$$\mathbf{y}_{n+1} - \mathbf{y}_n = h\Psi(\mathbf{y}_n, \mathbf{y}_{n+1}) , \quad (3.1)$$

where  $\Psi(\mathbf{y}, \mathbf{y}) = F(\mathbf{y})$  and  $\dot{\mathbf{y}} = F(\mathbf{y})$  is equation to be solved. This class contains, among others, the following numerical schemes:

- explicit Euler scheme (EEU):  $\Psi(\mathbf{y}_n, \mathbf{y}_{n+1}) = F(\mathbf{y}_n)$  ,
- implicit Euler scheme (IEU):  $\Psi(\mathbf{y}_n, \mathbf{y}_{n+1}) = F(\mathbf{y}_{n+1})$  ,
- implicit midpoint rule (IMP):  $\Psi(\mathbf{y}_n, \mathbf{y}_{n+1}) = F(\frac{\mathbf{y}_n + \mathbf{y}_{n+1}}{2})$  ,
- trapezoidal rule (TR):  $\Psi(\mathbf{y}_n, \mathbf{y}_{n+1}) = \frac{1}{2} (F(\mathbf{y}_n) + F(\mathbf{y}_{n+1}))$  .

A natural non-standard modification of (3.1) can be obtained by replacing  $h$  by a matrix  $\delta$ . Thus we consider the following family of modified numerical schemes:

$$\mathbf{y}_{n+1} - \mathbf{y}_n = \delta(\bar{\mathbf{y}}_n, h_n)\Psi(\mathbf{y}_n, \mathbf{y}_{n+1}) , \quad (3.2)$$

where  $\bar{\mathbf{y}}_n$  is described in Definition 2.4 and  $\delta(\bar{\mathbf{y}}_n, h_n)$  is a matrix-valued function. We assume consistency conditions:

$$\lim_{h \rightarrow 0} \frac{\delta(\bar{\mathbf{y}}_n, h)}{h} = I, \quad \Psi(\mathbf{y}, \mathbf{y}) = F(\mathbf{y}), \quad \Psi_1(\mathbf{y}, \mathbf{y}) + \Psi_2(\mathbf{y}, \mathbf{y}) = F'(\mathbf{y}), \quad (3.3)$$

where  $I$  is  $m \times m$  identity matrix,  $\Psi_1, \Psi_2$  are partial Fréchet derivative of  $\Psi$  with respect to the first and second vector variable, respectively (thus  $\Psi_1, \Psi_2$  are matrices). We also denote

$$\bar{\Psi} = \Psi(\bar{\mathbf{y}}_n, \bar{\mathbf{y}}_n) , \quad \bar{\Psi}_1 = \Psi_1(\bar{\mathbf{y}}_n, \bar{\mathbf{y}}_n) , \quad \bar{\Psi}_2 = \Psi_2(\bar{\mathbf{y}}_n, \bar{\mathbf{y}}_n) . \quad (3.4)$$

**Theorem 3.1.** *Numerical scheme (3.2), where  $\bar{\mathbf{y}}_n$  is any sequence described in Definition 2.4 and  $\Psi$  satisfies (3.3), is locally exact for*

$$\delta(\bar{\mathbf{y}}_n, h_n) = h_n \varphi_1(h_n F'(\bar{\mathbf{y}}_n)) (I + h_n \bar{\Psi}_2 \varphi_1(h_n F'(\bar{\mathbf{y}}_n)))^{-1} . \quad (3.5)$$

*Proof:* The exact discretization of the linearization of equation  $\dot{\mathbf{y}} = F(\mathbf{y})$  is given by (2.8). The linearization of the scheme (3.2) (at  $\bar{\mathbf{y}}_n$ ) reads

$$\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n = \boldsymbol{\delta}(\bar{\Psi}_1 \boldsymbol{\nu}_n + \bar{\Psi}_2 \boldsymbol{\nu}_{n+1} + \bar{\Psi}) , \quad (3.6)$$

where  $\boldsymbol{\nu}_n = \mathbf{y}_n - \bar{\mathbf{y}}_n$ ,  $\boldsymbol{\delta} = \boldsymbol{\delta}(\bar{\mathbf{y}}_n, h_n)$  and we use (3.4). Identifying (3.6) with (2.8) and assuming invertibility of  $I - \boldsymbol{\delta}\bar{\Psi}_2$  we get a system of two equations:

$$\begin{aligned} (I - \boldsymbol{\delta}\bar{\Psi}_2)^{-1}(I + \boldsymbol{\delta}\bar{\Psi}_1) &= e^{h_n F'} , \\ (I - \boldsymbol{\delta}\bar{\Psi}_2)^{-1}\boldsymbol{\delta}\bar{\Psi} &= h_n \varphi_1(h_n F') F , \end{aligned} \quad (3.7)$$

where  $F = F(\bar{\mathbf{y}}_n)$  and  $F' = F'(\bar{\mathbf{y}}_n)$ . Equations (3.7) imply

$$\begin{aligned} \boldsymbol{\delta}(\bar{\Psi}_1 + \bar{\Psi}_2 e^{h_n F'}) &= e^{h_n F'} - I , \\ \boldsymbol{\delta}(\bar{\Psi} + h_n \bar{\Psi}_2 \varphi_1(h_n F') F) &= h_n \varphi_1(h_n F') F . \end{aligned} \quad (3.8)$$

Using (3.3) and (3.4) (i.e.,  $\bar{\Psi} = F$ ,  $\bar{\Psi}_1 + \bar{\Psi}_2 = F'$ ), and replacing  $e^{h_n F'}$  by  $I + h_n F' \varphi_1(h_n F')$ , we rewrite (3.8) as follows

$$\begin{aligned} \boldsymbol{\delta}(I + h_n \bar{\Psi}_2 \varphi_1(h_n F')) F' &= h_n \varphi_1(h_n F') F' , \\ \boldsymbol{\delta}(I + h_n \bar{\Psi}_2 \varphi_1(h_n F')) F &= h_n \varphi_1(h_n F') F . \end{aligned} \quad (3.9)$$

One can easily see that  $\boldsymbol{\delta}$  given by (3.5) satisfies simultaneously both equations (3.9) (actually (3.5) is also necessary provided that  $F'$  is invertible).  $\square$

By straightforward calculation we can express matrix  $\boldsymbol{\delta}$  in the following (equivalent) form:

$$\boldsymbol{\delta}(\bar{\mathbf{y}}_n, h_n) = h_n \operatorname{tanhc} \frac{h_n F'}{2} \left( I + \frac{1}{2} h_n (\bar{\Psi}_2 - \bar{\Psi}_1) \operatorname{tanhc} \frac{h_n F'}{2} \right)^{-1} , \quad (3.10)$$

where  $\operatorname{tanhc}(z) \equiv z^{-1} \tanh(z)$  is analytic at  $z = 0$ . For small  $h_n$  we have

$$\boldsymbol{\delta}(\bar{\mathbf{y}}_n, h_n) = h_n I + \frac{1}{2} h_n^2 (\bar{\Psi}_1 - \bar{\Psi}_2) + O(h_n^3) .$$

Therefore  $\boldsymbol{\delta}$  exists for sufficiently small  $h_n$ . In other words, numerical scheme (3.2) admits a locally exact modification for sufficiently small  $h_n$ .

## Locally exact explicit Euler scheme

Specializing  $\Psi(\mathbf{y}_n, \mathbf{y}_{n+1}) = F(\mathbf{y}_n)$  we obtain a class of locally exact modifications of the explicit Euler scheme

$$\mathbf{x}_{n+1} = \mathbf{x}_n + (e^{h_n F'(\bar{\mathbf{x}})} - I)(F'(\bar{\mathbf{x}}))^{-1} F(\mathbf{x}_n) . \quad (3.11)$$

In this case the choice  $\bar{\mathbf{x}} = \mathbf{x}_n$  seems to be most natural.

**Proposition 3.2.** *Locally exact modification of the explicit Euler scheme (EEU-LEX) is given by*

$$\mathbf{x}_{n+1} = \mathbf{x}_n + (e^{h_n F'(\mathbf{x}_n)} - I)(F'(\mathbf{x}_n))^{-1} F(\mathbf{x}_n) . \quad (3.12)$$

Note that (3.12) coincides with (2.11).

## Locally exact implicit Euler schemes

For  $\Psi(\mathbf{y}_n, \mathbf{y}_{n+1}) = F(\mathbf{y}_{n+1})$  we get

$$\mathbf{x}_{n+1} = \mathbf{x}_n + (I - e^{-h_n F'(\bar{\mathbf{x}})})(F'(\bar{\mathbf{x}}))^{-1} F(\mathbf{x}_{n+1}) , \quad (3.13)$$

or, in an equivalent way,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h_n \varphi_1(-h_n F'(\bar{\mathbf{x}})) F(\mathbf{x}_{n+1}) . \quad (3.14)$$

In this case it is not clear what is the most natural identification. We can choose either  $\bar{\mathbf{x}} = \mathbf{x}_n$ , or  $\bar{\mathbf{x}} = \mathbf{x}_{n+1}$ .

**Proposition 3.3.** *Locally exact modification of the implicit Euler scheme is given either by*

$$\mathbf{x}_{n+1} = \mathbf{x}_n + (I - e^{-h_n F'(\mathbf{x}_n)})(F'(\mathbf{x}_n))^{-1} F(\mathbf{x}_{n+1}) , \quad (3.15)$$

*which will be called IEU-LEX, or by*

$$\mathbf{x}_{n+1} = \mathbf{x}_n + (I - e^{-h_n F'(\mathbf{x}_{n+1})})(F'(\mathbf{x}_{n+1}))^{-1} F(\mathbf{x}_{n+1}) , \quad (3.16)$$

*called IEU-ILEX.*

Both locally exact implicit Euler schemes are of second order. IEU-ILEX is a little bit more accurate, but IEU-LEX is more effective when computational costs are taken into account (see Section 7).



### Locally exact implicit midpoint rule

We take  $\Psi(\mathbf{y}_n, \mathbf{y}_{n+1}) = F(\frac{\mathbf{y}_n + \mathbf{y}_{n+1}}{2})$ . Then, using (3.10), we get

$$\delta_n = 2(F'(\bar{\mathbf{x}}))^{-1} \tanh\left(\frac{h_n F'(\bar{\mathbf{x}})}{2}\right). \quad (3.17)$$

Thus locally exact modification of the implicit midpoint rule is given by

$$\mathbf{x}_{n+1} - \mathbf{x}_n = h_n \left( \tanh\left(\frac{h_n F'(\bar{\mathbf{x}})}{2}\right) F\left(\frac{\mathbf{x}_{n+1} + \mathbf{x}_n}{2}\right) \right) \quad (3.18)$$

The most natural choice of  $\bar{\mathbf{x}}$  seems to be at the midpoint,  $\bar{\mathbf{x}} = \frac{1}{2}(\mathbf{x}_n + \mathbf{x}_{n+1})$ . The obtained scheme will be called IMP-SLEX. However, in order to diminish the computation cost, the choice  $\bar{\mathbf{x}} = \mathbf{x}_n$ , i.e., IMP-LEX, can also be considered (because then Jacobian  $F'(\bar{\mathbf{x}})$  is evaluated outside iteration loops).

### Locally exact trapezoidal rule

Taking into account  $\Psi(\mathbf{y}_n, \mathbf{y}_{n+1}) = \frac{1}{2}(F(\mathbf{y}_n) + F(\mathbf{y}_{n+1}))$  and (3.10), we get (3.17), as well. Thus locally exact modification of the trapezoidal rule reads

$$\mathbf{x}_{n+1} - \mathbf{x}_n = h_n \left( \tanh\left(\frac{h_n F'(\bar{\mathbf{x}})}{2}\right) \frac{F(\mathbf{x}_{n+1}) + F(\mathbf{x}_n)}{2} \right). \quad (3.19)$$

There are two natural choices of  $\bar{\mathbf{x}}$ . Either (in order to minimize the computational costs) we can take  $\bar{\mathbf{x}} = \mathbf{x}_n$ , obtaining TR-LEX, or (in order to get time-reversible scheme TR-SLEX) we can take  $\bar{\mathbf{x}} = \frac{1}{2}(\mathbf{x}_n + \mathbf{x}_{n+1})$ .

## 4 Locally exact discrete gradient schemes for multidimensional Hamiltonian systems

In this section we construct energy-preserving locally exact discrete gradient schemes for arbitrary multidimensional Hamiltonian systems in canonical coordinates:

$$\dot{x}^k = \frac{\partial H}{\partial p^k}, \quad \dot{p}^k = -\frac{\partial H}{\partial x^k}. \quad (4.1)$$

where  $k = 1, \dots, m$ . We obtain two different numerical schemes using either symmetric discrete gradient or coordinate increment discrete gradient.

## 4.1 Linearization of Hamiltonian systems

We denote  $\mathbf{x} = (x^1, \dots, x^m)^T$ ,  $\mathbf{p} = (p^1, \dots, p^m)$ , etc. The linearization of (4.1) around  $\bar{\mathbf{x}}, \bar{\mathbf{p}}$  is given by:

$$\begin{aligned}\dot{\boldsymbol{\xi}} &= H_{\mathbf{p}} + H_{\mathbf{p}\mathbf{x}}\boldsymbol{\xi} + H_{\mathbf{p}\mathbf{p}}\boldsymbol{\eta} , \\ \dot{\boldsymbol{\eta}} &= -H_{\mathbf{x}} - H_{\mathbf{x}\mathbf{x}}\boldsymbol{\xi} - H_{\mathbf{x}\mathbf{p}}\boldsymbol{\eta} ,\end{aligned}\tag{4.2}$$

where  $\boldsymbol{\xi} = \mathbf{x} - \bar{\mathbf{x}}$ ,  $\boldsymbol{\eta} = \mathbf{p} - \bar{\mathbf{p}}$ ,  $H_{\mathbf{p}}$  is a vector with components  $\frac{\partial H}{\partial p^k}$ ,  $H_{\mathbf{p}\mathbf{x}}$  is a matrix with coefficients  $\frac{\partial^2 H}{\partial p^j \partial x^k}$ , etc., and derivatives of  $H$  are evaluated at  $\bar{\mathbf{x}}, \bar{\mathbf{p}}$ . Note that  $m \times m$  matrices  $H_{\mathbf{x}\mathbf{x}}$ ,  $H_{\mathbf{x}\mathbf{p}}$ ,  $H_{\mathbf{p}\mathbf{x}}$ ,  $H_{\mathbf{p}\mathbf{p}}$  satisfy

$$H_{\mathbf{p}\mathbf{p}}^T = H_{\mathbf{p}\mathbf{p}} , \quad H_{\mathbf{x}\mathbf{x}}^T = H_{\mathbf{x}\mathbf{x}} , \quad H_{\mathbf{x}\mathbf{p}}^T = H_{\mathbf{p}\mathbf{x}}\tag{4.3}$$

Equations (4.2) can be rewritten also as

$$\frac{d}{dt} \begin{pmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{pmatrix} = F' \begin{pmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{pmatrix} + F\tag{4.4}$$

where

$$F = \begin{pmatrix} H_{\mathbf{p}} \\ -H_{\mathbf{x}} \end{pmatrix} , \quad F' = \begin{pmatrix} H_{\mathbf{p}\mathbf{x}} & H_{\mathbf{p}\mathbf{p}} \\ -H_{\mathbf{x}\mathbf{x}} & -H_{\mathbf{x}\mathbf{p}} \end{pmatrix} .\tag{4.5}$$

**Corollary 4.1.** *The exact discretization of linearized Hamiltonian equations (4.4) is given by*

$$\begin{pmatrix} \boldsymbol{\xi}_{n+1} \\ \boldsymbol{\eta}_{n+1} \end{pmatrix} = e^{h_n F'} \begin{pmatrix} \boldsymbol{\xi}_n \\ \boldsymbol{\eta}_n \end{pmatrix} + h_n \varphi_1(h_n F') F .\tag{4.6}$$

*Proof:* The exact discretization of (4.4) is given by (4.6) which follows immediately from Proposition 2.5.  $\square$

## 4.2 Discrete gradients in the multidimensional case

Considering multidimensional Hamiltonian systems we will denote

$$\mathbf{y} := \begin{pmatrix} \mathbf{x} \\ \mathbf{p} \end{pmatrix} , \quad \mathbf{y}_n := \begin{pmatrix} \mathbf{x}_n \\ \mathbf{p}_n \end{pmatrix} ,\tag{4.7}$$

where  $\mathbf{x} \in \mathbb{R}^m$ ,  $\mathbf{p} \in \mathbb{R}^m$ ,  $\mathbf{y} \in \mathbb{R}^{2m}$ , etc. In other words,

$$y^1 = x^1, \quad y^2 = x^2, \quad \dots \quad y^m = x^m, \quad y^{m+1} = p^1, \quad \dots \quad y^{2m} = p^m.$$

A discrete gradient, denoted by  $\bar{\nabla}H(\mathbf{y}_n, \mathbf{y}_{n+1})$ ,

$$\bar{\nabla}H = \left( \frac{\Delta H}{\Delta y^1}, \frac{\Delta H}{\Delta y^2}, \dots, \frac{\Delta H}{\Delta y^{2m}} \right) \equiv \left( \frac{\Delta H}{\Delta \mathbf{x}}, \frac{\Delta H}{\Delta \mathbf{p}} \right) \quad (4.8)$$

is defined as an  $\mathbb{R}^{2m}$ -valued function of  $\mathbf{y}_n, \mathbf{y}_{n+1}$  such that [12, 22]:

$$\sum_{k=1}^{2m} \frac{\Delta H}{\Delta y^k} (y_{n+1}^k - y_n^k) = H(\mathbf{y}_{n+1}) - H(\mathbf{y}_n), \quad (4.9)$$

$$\bar{\nabla}H(\mathbf{y}, \mathbf{y}) = (H_{\mathbf{x}}, H_{\mathbf{p}}),$$

where  $H_{\mathbf{x}}, H_{\mathbf{p}}$  are evaluated at  $\mathbf{y} = (\mathbf{x}, \mathbf{p})$  and we define

$$\bar{\nabla}H(\mathbf{y}, \mathbf{y}) = \lim_{\tilde{\mathbf{y}} \rightarrow \mathbf{y}} \bar{\nabla}H(\mathbf{y}, \tilde{\mathbf{y}}) \quad (4.10)$$

when necessary. It is well known (see, e.g., [12, 13, 20, 22]) that any discrete gradient defines an energy-preserving numerical scheme

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h \bar{\nabla}H. \quad (4.11)$$

Discrete gradients are non-unique, compare [12, 19, 22]. Here we confine ourselves to the simplest discrete gradients. We consider coordinate increment discrete gradient, first proposed by Itoh and Abe [19], and its symmetrization (see below). The coordinate increment discrete gradient is defined by:

$$\begin{aligned} \frac{\Delta H}{\Delta y^1} &= \frac{H(y_{n+1}^1, y_n^2, y_n^3, \dots, y_n^{2m}) - H(y_n^1, y_n^2, y_n^3, \dots, y_n^{2m})}{y_{n+1}^1 - y_n^1}, \\ \frac{\Delta H}{\Delta y^2} &= \frac{H(y_{n+1}^1, y_{n+1}^2, y_n^3, \dots, y_n^{2m}) - H(y_{n+1}^1, y_n^2, \dots, y_n^{2m})}{y_{n+1}^2 - y_n^2}, \\ &\dots \dots \dots \end{aligned} \quad (4.12)$$

$$\frac{\Delta H}{\Delta y^{2m}} = \frac{H(y_{n+1}^1, y_{n+1}^2, \dots, y_{n+1}^{2m}) - H(y_{n+1}^1, y_{n+1}^2, \dots, y_n^{2m})}{y_{n+1}^{2m} - y_n^{2m}}.$$

where, to fix our attention,  $\mathbf{y}$  is defined by (4.7). The corresponding numerical scheme (4.11) will be called GR-IA. In fact, we may identify with  $\mathbf{y}$  any permutation of  $2m$  components  $x^k, p^j$ . Thus we have  $(2m)!$  discrete gradients of this type (some of them may happen to be identical).

Having any discrete gradient we can easily obtain the related symmetric discrete gradient

$$\bar{\nabla}_s H(\mathbf{y}_n, \mathbf{y}_{n+1}) = \frac{1}{2} (\bar{\nabla} H(\mathbf{y}_n, \mathbf{y}_{n+1}) + \bar{\nabla} H(\mathbf{y}_{n+1}, \mathbf{y}_n)) . \quad (4.13)$$

One can easily verify that  $\bar{\nabla}_s H$  satisfies conditions (4.9) provided that they are satisfied by  $\bar{\nabla} H$ . Discrete gradient scheme obtained by symmetrization (4.13) from GR-IA will be called GR-SYM.

### 4.3 Linearization of discrete gradients

In order to construct locally exact modifications we need to linearize discrete gradients defined by (4.12) and (4.13).

**Lemma 4.2.** *Linearization of the coordinate increment discrete gradient (4.12) around  $\bar{\mathbf{y}}$  yields*

$$\bar{\nabla} H(\mathbf{y}_n, \mathbf{y}_{n+1}) \approx H_{\mathbf{y}} + \frac{1}{2} (A\boldsymbol{\nu}_{n+1} + B\boldsymbol{\nu}_n), \quad (4.14)$$

where we denoted  $\boldsymbol{\nu}_n = \mathbf{y}_n - \bar{\mathbf{y}}$ , and

$$A = \begin{pmatrix} \frac{1}{2}H_{y^1y^1} & 0 & \dots & 0 & 0 \\ H_{y^2y^1} & \frac{1}{2}H_{y^2y^2} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ H_{y^{2m-1}y^1} & H_{y^{2m-1}y^2} & \dots & \frac{1}{2}H_{y^{2m-1}y^{2m-1}} & 0 \\ H_{y^{2m}y^1} & H_{y^{2m}y^2} & \dots & H_{y^{2m}y^{2m-1}} & \frac{1}{2}H_{y^{2m}y^{2m}} \end{pmatrix}, \quad (4.15)$$

$$B = \begin{pmatrix} \frac{1}{2}H_{y^1y^1} & H_{y^1y^2} & \dots & H_{y^1y^{2m-1}} & H_{y^1y^{2m}} \\ 0 & \frac{1}{2}H_{y^2y^2} & \dots & H_{y^2y^{2m-1}} & H_{y^2y^{2m}} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \frac{1}{2}H_{y^{2m-1}y^{2m-1}} & H_{y^{2m-1}y^{2m}} \\ 0 & 0 & \dots & 0 & \frac{1}{2}H_{y^{2m}y^{2m}} \end{pmatrix},$$

where  $H_{y^jy^k} := \frac{\partial^2 H}{\partial y^j \partial y^k}$ .

*Proof:* We denote  $\hat{\mathbf{y}}_n^j = (y_{n+1}^1, \dots, y_{n+1}^j, y_n^{j+1}, \dots, y_n^{2m})^T$ . In particular,  $\hat{\mathbf{y}}_n^0 = \mathbf{y}_n$  and  $\hat{\mathbf{y}}_n^{2m} = \mathbf{y}_{n+1}$ . Expanding  $H(\hat{\mathbf{y}}_n^j)$  around  $\hat{\mathbf{y}}_n^{j-1}$ , we get

$$H(\hat{\mathbf{y}}_n^j) = H(\hat{\mathbf{y}}_n^{j-1}) + H_{y^j}(\hat{\mathbf{y}}_n^{j-1})(y_{n+1}^j - y_n^j) + \frac{1}{2}H_{y^j y^j}(\hat{\mathbf{y}}_n^{j-1})(y_{n+1}^j - y_n^j)^2 + \dots \quad (4.16)$$

Hence

$$\frac{\Delta H}{\Delta y^j} = \frac{H(\hat{\mathbf{y}}_n^j) - H(\hat{\mathbf{y}}_n^{j-1})}{y_{n+1}^j - y_n^j} = H_{y^j}(\hat{\mathbf{y}}_n^{j-1}) + \frac{1}{2}H_{y^j y^j}(\hat{\mathbf{y}}_n^{j-1})(y_{n+1}^j - y_n^j) + \dots \quad (4.17)$$

Then, from the definition of  $\boldsymbol{\nu}_n$ , we have

$$\begin{aligned} y_{n+1}^j - y_n^j &= \nu_{n+1}^j - \nu_n^j, \\ \hat{\mathbf{y}}_n^{j-1} &= \bar{\mathbf{y}} + \left( \nu_{n+1}^1, \dots, \nu_{n+1}^{j-1}, \nu_n^j, \dots, \nu_n^{2m} \right), \end{aligned} \quad (4.18)$$

and, taking it into account, we rewrite (4.17) as

$$\frac{\Delta H}{\Delta y^j} = H_{y^j}(\bar{\mathbf{y}}) + \sum_{k=1}^{j-1} H_{y^j y^k}(\bar{\mathbf{y}}) \nu_{n+1}^k + \sum_{k=j}^{2m} H_{y^j y^k}(\bar{\mathbf{y}}) \nu_n^k + \frac{1}{2}H_{y^j y^j}(\bar{\mathbf{y}})(\nu_{n+1}^j - \nu_n^j) + \dots \quad (4.19)$$

which is equivalent to (4.14), (4.15).  $\square$

**Lemma 4.3.** *Linearization of the symmetric discrete gradient yields*

$$\bar{\nabla}_s H(\mathbf{y}_n, \mathbf{y}_{n+1}) \approx H_{\mathbf{y}} + \frac{1}{2}H_{\mathbf{y}\mathbf{y}}(\boldsymbol{\nu}_n + \boldsymbol{\nu}_{n+1}), \quad (4.20)$$

where  $H_{\mathbf{y}\mathbf{y}}$  is the Hessian matrix of  $H$ , evaluated at  $\bar{\mathbf{y}}$ .

*Proof:* We observe that  $A + B = H_{\mathbf{y}\mathbf{y}}$  and then we use (4.13) and (4.14).  $\square$

## 4.4 Conservative properties of modified discrete gradients

In the one-dimensional case locally exact modifications are clearly energy-preserving, compare [9, 10]. In the general case, conservative properties are less obvious. In this section we present several useful results.

**Lemma 4.4.** *We assume that a  $2m \times 2m$  matrix  $\Lambda$  (depending on  $h$  and, possibly, on other variables) is skew-symmetric (i.e.,  $\Lambda^T = -\Lambda$ ) and*

$$\lim_{h \rightarrow 0} \frac{\Lambda}{h} = S, \quad S = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}. \quad (4.21)$$

*Then, the numerical scheme*

$$\mathbf{y}_{n+1} - \mathbf{y}_n = \Lambda \bar{\nabla} H, \quad (4.22)$$

*where  $\mathbf{y}_n$  is defined by (4.7) and  $\bar{\nabla} H$  satisfies (4.9), is a consistent integrator for (4.1) preserving the energy integral up to round-off error.*

*Proof:* The consistency follows immediately from (4.21). The energy preservation can be shown in the standard way. Using the standard scalar product in  $\mathbb{R}^{2m}$ , we multiply both sides of (4.22) by  $\bar{\nabla} H$

$$\langle \bar{\nabla} H | \mathbf{y}_{n+1} - \mathbf{y}_n \rangle = \langle \bar{\nabla} H | \Lambda \bar{\nabla} H \rangle. \quad (4.23)$$

By virtue of (4.9) the left-hand side equals  $H(\mathbf{y}_{n+1}) - H(\mathbf{y}_n)$ . The right hand side vanishes due to the skew symmetry of  $\Lambda$ . Hence  $H(\mathbf{y}_{n+1}) = H(\mathbf{y}_n)$ .  $\square$

**Remark 4.5.** *Scheme (4.22) for  $\Lambda = hS$  becomes a standard discrete gradient scheme.*

**Lemma 4.6.** *We assume that  $2m \times 2m$  matrix  $\boldsymbol{\theta}$  is of the following form*

$$\boldsymbol{\theta} = \begin{pmatrix} \boldsymbol{\delta} & -\boldsymbol{\sigma} \\ \boldsymbol{\rho} & \boldsymbol{\delta}^T \end{pmatrix}, \quad \boldsymbol{\rho}^T = -\boldsymbol{\rho}, \quad \boldsymbol{\sigma}^T = -\boldsymbol{\sigma}, \quad \lim_{h \rightarrow 0} \frac{\boldsymbol{\theta}}{h} = I \quad (4.24)$$

*(where  $\boldsymbol{\delta}, \boldsymbol{\sigma}, \boldsymbol{\rho}$  are  $m \times m$  matrices) and  $\bar{\nabla} H$  is (any) discrete gradient. Then, the numerical scheme given by*

$$\mathbf{y}_{n+1} - \mathbf{y}_n = \boldsymbol{\theta} S \bar{\nabla} H \quad (4.25)$$

*preserves the energy integral exactly, i.e.,  $H(\mathbf{x}_{n+1}, \mathbf{y}_{n+1}) = H(\mathbf{x}_n, \mathbf{y}_n)$ .*

*Proof:* We observe that  $\boldsymbol{\theta} S$  is skew-symmetric, and then we use Lemma 4.4.  $\square$

We easily see that (4.25) reduces to the standard discrete gradient scheme when we take  $\boldsymbol{\theta} = h_n I$  (i.e.,  $\boldsymbol{\theta}$  is proportional to the unit matrix).

**Lemma 4.7.** *If  $\boldsymbol{\theta}$  is of the form (4.24) and  $z \mapsto f(z)$  is any analytic function, then  $f(\boldsymbol{\theta})$  is also of the form (4.24).*

*Proof:* First, we will show that the conditions (4.24) are equivalent to

$$\boldsymbol{\theta}^T = S^{-1}\boldsymbol{\theta}S . \quad (4.26)$$

Indeed, assuming a general form of  $\boldsymbol{\theta}$ , e.g.,  $\boldsymbol{\theta} = \begin{pmatrix} \boldsymbol{\delta} & -\boldsymbol{\sigma} \\ \boldsymbol{\rho} & \boldsymbol{\gamma} \end{pmatrix}$  we see that the constraint (4.26) is equivalent to  $\boldsymbol{\gamma} = \boldsymbol{\delta}^T$ ,  $\boldsymbol{\rho}^T = -\boldsymbol{\rho}$  and  $\boldsymbol{\sigma}^T = -\boldsymbol{\sigma}$ . Then the proof is straightforward. Assuming  $f(z) = \sum_{k=1}^{\infty} a_k z^k$ , we obtain

$$\left( \sum_{k=1}^{\infty} a_k \boldsymbol{\theta}^k \right)^T = \sum_{k=1}^{\infty} a_k (S^{-1}\boldsymbol{\theta}S)^k = S^{-1} \left( \sum_{k=1}^{\infty} a_k \boldsymbol{\theta}^k \right) S , \quad (4.27)$$

i.e.,  $f(\boldsymbol{\theta})^T = S^{-1}f(\boldsymbol{\theta})S$ . □

**Corollary 4.8.** *If  $\boldsymbol{\theta}^T = S^{-1}\boldsymbol{\theta}S$  and  $f$  is an analytic function, then the scheme  $\mathbf{y}_{n+1} - \mathbf{y}_n = f(\boldsymbol{\theta})S\bar{\nabla}H$  exactly preserves the energy integral  $H$ .*

## 4.5 Locally exact symmetric discrete gradient scheme

We begin with the symmetric case because the coordinate increment discrete gradient case is more difficult. In the symmetric case a locally exact modification is derived similarly as in the one-dimensional case.

**Proposition 4.9.** *The following modification of a symmetric discrete gradient scheme is locally exact at  $\bar{\mathbf{y}}$ :*

$$\mathbf{y}_{n+1} - \mathbf{y}_n = \boldsymbol{\theta}_n S \bar{\nabla}_s H , \quad (4.28)$$

where

$$\boldsymbol{\theta}_n = 2(F')^{-1} \tanh \frac{h_n F'}{2} , \quad (4.29)$$

and  $F'$ , given by (4.5), is evaluated at  $\mathbf{y} = \bar{\mathbf{y}}$ .

*Proof:* We are going to derive (4.29), assuming that  $\boldsymbol{\theta}_n$  depends on  $\bar{\mathbf{y}}$  and  $h$ . By virtue of Lemma 4.3 the linearization of (4.28) is given by

$$\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n = \boldsymbol{\theta}_n S \left( H_{\mathbf{y}} + \frac{1}{2} H_{\mathbf{y}\mathbf{y}} (\boldsymbol{\nu}_{n+1} + \boldsymbol{\nu}_n) \right). \quad (4.30)$$

Taking into account (4.5) we transform (4.30) into

$$\left( I - \frac{1}{2} \boldsymbol{\theta}_n F' \right) \boldsymbol{\nu}_{n+1} = \left( I + \frac{1}{2} \boldsymbol{\theta}_n F' \right) \boldsymbol{\nu}_n + \boldsymbol{\theta}_n F. \quad (4.31)$$

The scheme (4.28) is locally exact if and only if (4.31) coincides with (4.6). Therefore, we require that

$$\left( I - \frac{1}{2} \boldsymbol{\theta}_n F' \right) e^{h_n F'} = I + \frac{1}{2} \boldsymbol{\theta}_n F', \quad (4.32)$$

$$\left( I - \frac{1}{2} \boldsymbol{\theta}_n F' \right) \left( e^{h_n F'} - I \right) (F')^{-1} F = \boldsymbol{\theta}_n F. \quad (4.33)$$

From equation (4.32) we can compute  $\boldsymbol{\theta}_n$  which yields (4.29). Equation (4.33) is automatically satisfied provided that (4.32) holds.  $\square$

Taking the symmetrization of the Itoh-Abe discrete gradient (compare (4.13)), modifying it according to (4.28) and choosing  $\bar{\mathbf{y}} = \mathbf{y}_n$  we get a scheme called GR-SYM-LEX, while for  $\bar{\mathbf{y}} = \frac{1}{2} (\mathbf{y}_n + \mathbf{y}_{n+1})$  we obtain GR-SYM-SLEX.

**Proposition 4.10.** *The numerical scheme (4.28) with  $\boldsymbol{\theta}_n$  given by (4.29) is energy-preserving.*

*Proof:* We have  $F' = S H_{\mathbf{y}\mathbf{y}}$ . Therefore,  $(F')^T = -H_{\mathbf{y}\mathbf{y}} S = -S^{-1} F' S$ , and, as a consequence

$$((F')^2)^T = S^{-1} (F')^2 S. \quad (4.34)$$

It means that  $(F')^2$  has the form (4.24). The formula (4.29) expresses  $\boldsymbol{\theta}_n$  as an analytic function of  $(F')^2$ . Finally, we use Lemma 4.7.  $\square$

In the one-dimensional case  $(F')^2$  is proportional to the unit matrix which essentially simplifies arguments presented in this section, see [9, 10].



## 4.6 Locally exact coordinate increment discrete gradient scheme

The symmetric form of the discrete gradient leads to a simple form of the locally exact modification. It turns out, however, that starting from the simplest form of the discrete gradient, namely coordinate increment discrete gradient GR-IA, we also succeed in deriving corresponding locally exact modifications.

**Proposition 4.11.** *The following modification of the coordinate increment discrete gradient scheme is locally exact at  $\bar{\mathbf{y}}$ :*

$$\mathbf{y}_{n+1} - \mathbf{y}_n = \boldsymbol{\theta}_n S \bar{\nabla} H, \quad (4.35)$$

where  $\bar{\nabla} H$  is given by (4.12),

$$\theta_n = 2 \left( SR + F' \coth \frac{h_n F'}{2} \right)^{-1}, \quad (4.36)$$

$F'$  is given by (4.5) (i.e.,  $F' = SH_{\mathbf{y}\mathbf{y}}$ ), and, finally  $R = A - B$ , i.e.,

$$R = \begin{pmatrix} 0 & -H_{y^1 y^2} & \dots & -H_{y^1 y^{2m-1}} & -H_{y^1 y^{2m}} \\ H_{y^2 y^1} & 0 & \dots & -H_{y^2 y^{2m-1}} & -H_{y^2 y^{2m}} \\ \dots & \dots & \dots & \dots & \dots \\ H_{y^{2m-1} y^1} & H_{y^{2m-1} y^2} & \dots & 0 & -H_{y^{2m-1} y^{2m}} \\ H_{y^{2m} y^1} & H_{y^{2m} y^2} & \dots & H_{y^{2m} y^{2m-1}} & 0 \end{pmatrix}. \quad (4.37)$$

$F'$  and  $R$  are evaluated at  $\mathbf{y} = \bar{\mathbf{y}}$ .

*Proof:* We are going to derive (4.36), assuming that  $\boldsymbol{\theta}_n$  depends on  $\bar{\mathbf{y}}$  and  $h$ . By virtue of Lemma 4.2 the linearization of (4.35) is given by

$$\boldsymbol{\nu}_{n+1} - \boldsymbol{\nu}_n = \boldsymbol{\theta}_n S(A\boldsymbol{\nu}_{n+1} + B\boldsymbol{\nu}_n) + \boldsymbol{\theta}_n SH_{\mathbf{y}}. \quad (4.38)$$

Hence, taking into account that  $SH_{\mathbf{y}} = F$ ,

$$(I - \boldsymbol{\theta}_n SA)\boldsymbol{\nu}_{n+1} = (I + \boldsymbol{\theta}_n SB)\boldsymbol{\nu}_n + \boldsymbol{\theta}_n F. \quad (4.39)$$

The scheme (4.35) is locally exact if and only if (4.39) coincides with (4.6). Therefore, we require that

$$(I - \boldsymbol{\theta}_n SA) e^{h_n F'} = I + \boldsymbol{\theta}_n SB, \quad (4.40)$$

$$(I - \boldsymbol{\theta}_n S A) \left( e^{h_n F'} - I \right) (F')^{-1} F = \boldsymbol{\theta}_n F . \quad (4.41)$$

Inserting (4.40) into (4.41) we get

$$\boldsymbol{\theta}_n S (B + A) (F')^{-1} F = \boldsymbol{\theta}_n F , \quad (4.42)$$

which is identically satisfied by virtue of  $A + B = H_{\mathbf{y}\mathbf{y}} = S^{-1} F'$ , compare (4.15). The remaining equation, (4.40), defines  $\boldsymbol{\theta}_n$ :

$$\boldsymbol{\theta}_n \left( S A e^{h_n F'} + S B \right) = e^{h_n F'} - I . \quad (4.43)$$

In order to simplify (4.43) we introduce  $R = A - B$  ( $R$  is antisymmetric because  $B = A^T$ ). Then, taking into account  $SA + SB = F'$ , we get

$$SA = \frac{1}{2} F' + \frac{1}{2} SR , \quad SB = \frac{1}{2} F' - \frac{1}{2} SR . \quad (4.44)$$

Substituting it into (4.43) we complete the proof.  $\square$

Choosing  $\bar{\mathbf{y}} = \mathbf{y}_n$  we obtain GR-IA-LEX, while for  $\bar{\mathbf{y}} = \frac{1}{2} (\mathbf{y}_n + \mathbf{y}_{n+1})$  we get GR-IA-SLEX.

**Proposition 4.12.** *The numerical scheme (4.35) with  $\boldsymbol{\theta}_n$  given by (4.36) is energy-preserving, i.e.,  $H(\mathbf{y}_{n+1}) = H(\mathbf{y}_n)$ .*

*Proof:* We have

$$\boldsymbol{\theta}_n^T = 2 \left( (SR)^T + \left( F' \coth \frac{h_n F'}{2} \right)^T \right)^{-1} = S^{-1} \boldsymbol{\theta}_n S , \quad (4.45)$$

because

$$(SR)^T = (-R)(-S) = S^{-1} (SR) S \quad (4.46)$$

and, by virtue of Lemma 4.7,

$$\left( F' \coth \frac{h_n F'}{2} \right)^T = S^{-1} \left( F' \coth \frac{h_n F'}{2} \right) S , \quad (4.47)$$

where we took into account (4.34). Then, we use Corollary 4.8.  $\square$

Locally exact modification (4.35) for Hamiltonian  $H = T(\mathbf{p}) + V(\mathbf{x})$  with two degrees of freedom is determined by

$$\theta_n = h_n \tanh c \frac{h_n F'}{2} \left( I + \frac{1}{2} h_n S R \tanh c \frac{h_n F'}{2} \right)^{-1}, \quad (4.48)$$

where

$$R = \begin{pmatrix} 0 & -V_{,12} & 0 & 0 \\ V_{,12} & 0 & 0 & 0 \\ 0 & 0 & 0 & -T_{,12} \\ 0 & 0 & T_{,12} & 0 \end{pmatrix} \quad (4.49)$$

In numerical experiments we use  $T(\mathbf{p}) = \frac{1}{2}\mathbf{p}^2$  and a potential depending only on  $r = |\mathbf{x}|$ . In this case  $T_{,12} = 0$  and  $V_{,12} = \frac{x^1 x^2}{r} \left( \frac{1}{r} V_{,r} \right)_{,r}$ . After straightforward calculations we reduce (4.35) to:

$$\begin{aligned} \mathbf{x}_{n+1} - \mathbf{x}_n &= hD \frac{\mathbf{p}_{n+1} + \mathbf{p}_n}{2}, \\ \mathbf{p}_{n+1} - \mathbf{p}_n &= -hD \bar{\nabla} V - \frac{1}{2} h^2 (\det D) V_{,12} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \end{aligned} \quad (4.50)$$

where  $\Omega^2 = V_{,\mathbf{x}\mathbf{x}}$  and  $D = \tanh c \frac{h\Omega}{2}$ .

## 5 Order of considered numerical schemes

First of all, we recall the Taylor expansion of the exact solution of equation  $\dot{\mathbf{x}} = F(\mathbf{x})$  (compare, e.g., [2, 14]):

$$\mathbf{x}(t+h) = \mathbf{x} + hF + \frac{1}{2} h^2 F' F + \frac{1}{3!} h^3 (F''(F, F) + (F')^2 F) + \dots \quad (5.1)$$

where all terms on the right-hand side are evaluated at  $t$ . Note that  $F$  is a vector,  $F'$  is a matrix,  $F''$  is a vector-valued bilinear form, etc. Then,  $\mathbf{x}(t)$  and  $\mathbf{x}(t+h)$  will be identified with  $\mathbf{x}_n$  and  $\mathbf{x}_{n+1}$ , respectively. To simplify notation, in this section we replace  $h_n$  by  $h$ . Computing the order of numerical schemes

we use expansions

$$\begin{aligned}
F(\mathbf{x}_{n+1}) &= F + hF'b_1 + h^2 \left( F'b_2 + \frac{1}{2!}F''(b_1, b_1) \right) \\
&\quad + h^3 \left( F'b_3 + F''(b_1, b_2) + \frac{1}{3!}F'''(b_1, b_1, b_1) \right) + \dots \\
F\left(\frac{\mathbf{x}_n + \mathbf{x}_{n+1}}{2}\right) &= F + \frac{1}{2}hF'b_1 + h^2 \left( \frac{1}{2}F'b_2 + \frac{1}{8}F''(b_1, b_1) \right) \\
&\quad + h^3 \left( \frac{1}{2}F'b_3 + \frac{1}{4}F''(b_1, b_2) + \frac{1}{48}F'''(b_1, b_1, b_1) \right) + \dots
\end{aligned} \tag{5.2}$$

where  $b_k$  are coefficients of the Taylor expansion of  $\Phi(\mathbf{x}_n, h)$ , i.e.,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + b_1h + b_2h^2 + b_3h^3 + \dots \tag{5.3}$$

In the case of locally exact schemes we use also the following Taylor expansions

$$\begin{aligned}
\varphi_1(hF') &= I + \frac{1}{2}hF' + \frac{1}{3!}h^2(F')^2 + \frac{1}{4!}h^3(F')^3 + \dots \\
\tanhc \frac{h_n F'}{2} &= I - \frac{1}{12}h_n^2(F')^2 + \frac{1}{120}h_n^4(F')^4 + \dots
\end{aligned} \tag{5.4}$$

substituting (5.2) if  $F'$  is evaluated at  $\mathbf{x}_{n+1}$  or  $\frac{1}{2}(\mathbf{x}_n + \mathbf{x}_{n+1})$ . The final result of this analysis (expansions (5.3) for particular numerical schemes) is presented below.

### First order schemes

- Explicit (forward) Euler scheme, EEU,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hF. \tag{5.5}$$

- Implicit (backward) Euler scheme, IEU,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hF + h^2F'F + \dots \tag{5.6}$$

## Second order schemes

- Locally exact explicit Euler scheme ( $\bar{\mathbf{x}} = \mathbf{x}_n$ ), EEU-LEX,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hF + \frac{1}{2}h^2F'F + \frac{1}{6}h^3(F')^2F + \dots \quad (5.7)$$

- Locally exact implicit Euler scheme ( $\bar{\mathbf{x}} = \mathbf{x}_n$ ), IEU-LEX,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hF + \frac{1}{2}h^2F'F + h^3 \left( \frac{1}{6}(F')^2F + \frac{1}{2}F''(F, F) \right) + \dots \quad (5.8)$$

- Locally exact implicit Euler scheme ( $\bar{\mathbf{x}} = \mathbf{x}_{n+1}$ ), IEU-ILEX,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hF + \frac{1}{2}h^2F'F + h^3 \left( \frac{1}{6}(F')^2F + \frac{1}{4}F''(F, F) \right) + \dots \quad (5.9)$$

- Implicit midpoint rule, IMP,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hF + \frac{1}{2}h^2F'F + h^3 \left( \frac{1}{4}(F')^2F + \frac{1}{8}F''(F, F) \right) + \dots \quad (5.10)$$

- Locally exact implicit midpoint rules IMP-LEX, IMP-SLEX ( $h$ -expansions for  $\bar{\mathbf{x}} = \mathbf{x}_n$  and  $\bar{\mathbf{x}} = \frac{1}{2}(\mathbf{x}_n + \mathbf{x}_{n+1})$  are identical up to the third order):

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hF + \frac{1}{2}h^2F'F + h^3 \left( \frac{1}{6}(F')^2F + \frac{1}{8}F''(F, F) \right) + \dots \quad (5.11)$$

- Trapezoidal rule, TR,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hF + \frac{1}{2}h^2F'F + h^3 \left( \frac{1}{4}(F')^2F + \frac{1}{4}F''(F, F) \right) + \dots \quad (5.12)$$

- Locally exact trapezoidal rules TR-LEX, TR-SLEX ( $h$ -expansions for  $\bar{\mathbf{x}} = \frac{1}{2}(\mathbf{x}_n + \mathbf{x}_{n+1})$  and  $\bar{\mathbf{x}} = \mathbf{x}_n$  are identical up to the third order):

$$\mathbf{x}_{n+1} = \mathbf{x}_n + hF + \frac{1}{2}h^2F'F + h^3 \left( \frac{1}{6}(F')^2F + \frac{1}{4}F''(F, F) \right) + \dots \quad (5.13)$$

## Gradient schemes

Discussing gradient schemes we confine ourselves to Hamiltonians of the form  $H = T(\mathbf{p}) + V(\mathbf{x})$ . First, we expand a discrete gradient of  $V$  with respect to  $\Delta x^j$  ( $j$ th component of  $\mathbf{x}_{n+1} - \mathbf{x}_n$ , compare (4.8)):

$$\frac{\Delta V}{\Delta x^j} = V_{,x^j} + \frac{1}{2}A_{jk}\Delta x^k + \frac{1}{3!}B_{j\mu\nu}\Delta x^\mu\Delta x^\nu + \frac{1}{4!}C_{jk\mu\nu}\Delta x^k\Delta x^\mu\Delta x^\nu + \dots \quad (5.14)$$

where  $A_{jk}$ ,  $B_{j\mu\nu}$  and  $C_{jk\mu\nu}$  are  $\mathbf{x}_n$ -dependent coefficients. In particular:

- Itoh-Abe gradient , GR-IA,

$$A_{jj} = V_{,x^j x^j} , \quad A_{jk} = 0 \quad (j < k), \quad A_{jk} = 2V_{,x^j x^k} \quad (j > k), \quad (5.15)$$

- any symmetric discrete gradient, compare (4.13),

$$A_{jk} = V_{,x^j x^k} , \quad (5.16)$$

- symmetrization of the Itoh-Abe gradient, one degree of freedom,

$$A_{11} = V_{,xx} , \quad B_{111} = V_{,xxx} , \quad C_{1111} = V_{,xxxx} , \quad (5.17)$$

- symmetrization of the Itoh-Abe gradient (GR-SYM), two degrees of freedom,

$$\begin{aligned} B_{111} &= V_{,xxx} , \quad B_{112} = \frac{3}{4}V_{,xxy} , \quad B_{122} = \frac{3}{2}V_{,xyy} , \\ B_{222} &= V_{,yyy} , \quad B_{212} = \frac{3}{4}V_{,xyy} , \quad B_{211} = \frac{3}{2}V_{,xxy} . \end{aligned} \quad (5.18)$$

Having (5.14) we substitute expansion of  $\Delta x^j$  with respect to  $h$  in the considered discrete gradient scheme. As a results we obtain Taylor series for  $\mathbf{x}_{n+1}$  and comparing them with (5.1) we arrive at the following conclusions (assuming the generic case, because for some exceptional  $V$ , e.g.,  $V$  linear or quadratic in  $x^j$ , the order can be higher).

### First order schemes

- Discrete gradient schemes such that  $A_{jk} \neq V_{,x^j x^k}$ , in particular: the Itoh-Abe scheme GR-IA.

## Second order schemes

- Discrete gradient schemes such that  $A_{jk} = V_{,x^j x^k}$ , in particular: symmetric gradient schemes (including GR-SYM) and one-dimensional discrete gradient method.
- Locally exact ( $\bar{\mathbf{x}} = \mathbf{x}_n$ , i.e., LEX) modifications of symmetric discrete gradient schemes such that  $B_{j\mu\nu} \neq V_{,x^j x^\mu x^\nu}$ . In particular, locally exact modification of symmetrization of the Itoh-Abe discrete gradient scheme (GR-SYM-LEX), multidimensional case (i.e., two degrees of freedom, at least).
- Locally exact time reversible ( $\bar{\mathbf{x}} = \frac{1}{2}(\mathbf{x}_n + \mathbf{x}_{n+1})$ , i.e., SLEX) modifications of symmetric discrete gradient schemes such that  $B_{j\mu\nu} \neq V_{,x^j x^\mu x^\nu}$ . In particular, locally exact time reversible modification of symmetrization of the Itoh-Abe discrete gradient scheme (GR-SYM-SLEX), multidimensional case.
- Locally exact ( $\bar{\mathbf{x}} = \mathbf{x}_n$ ) modification of the Itoh-Abe discrete gradient scheme (GR-IA-LEX).
- Locally exact time reversible ( $\bar{\mathbf{x}} = \frac{1}{2}(\mathbf{x}_n + \mathbf{x}_{n+1})$ ) modification of the Itoh-Abe discrete gradient scheme (GR-IA-SLEX).

## Third order schemes

- Locally exact (LEX:  $\bar{\mathbf{x}} = \mathbf{x}_n$ ) modifications of symmetric discrete gradient schemes such that  $B_{j\mu\nu} = V_{,x^j x^\mu x^\nu}$ . In particular, GR-LEX in one-dimensional case, see [10].
- Locally exact time reversible ( $\bar{\mathbf{x}} = \frac{1}{2}(\mathbf{x}_n + \mathbf{x}_{n+1})$ , i.e., SLEX) modifications of symmetric discrete gradient schemes such that  $B_{j\mu\nu} = V_{,x^j x^\mu x^\nu}$  and  $C_{jk\mu\nu} \neq V_{,x^j x^k x^\mu x^\nu}$ .

## Fourth order schemes

- Locally exact time reversible (SLEX:  $\bar{\mathbf{x}} = \frac{1}{2}(\mathbf{x}_n + \mathbf{x}_{n+1})$ ) modifications of symmetric discrete gradient schemes such that  $B_{j\mu\nu} = V_{,x^j x^\mu x^\nu}$  and  $C_{jk\mu\nu} = V_{,x^j x^k x^\mu x^\nu}$ . In particular, GR-SLEX in one-dimensional case, see [10].

## 6 Linear stability

Locally exact numerical schemes have excellent qualitative behaviour around all fixed points of the considered system.

**Proposition 6.1.** *If the equation  $\dot{\mathbf{x}} = F(\mathbf{x})$  has a fixed point at  $\mathbf{x} = \bar{\mathbf{x}}$ , then all its locally exact discretizations have a fixed point at  $\mathbf{x}_n = \bar{\mathbf{x}}$ , as well.*

*Proof:* If  $F(\bar{\mathbf{x}}) = 0$ , then equation (2.10) becomes

$$\mathbf{x}_{n+1} - \bar{\mathbf{x}} = e^{h_n F'(\bar{\mathbf{x}})} (\mathbf{x}_n - \bar{\mathbf{x}}) . \quad (6.1)$$

We require that the scheme  $\mathbf{x}_{n+1} = \Phi(\mathbf{x}_n, h)$  is a locally exact discretization of  $F(\bar{\mathbf{x}}) = 0$ , i.e., its linearization, given by

$$\mathbf{x}_{n+1} = \Phi(\bar{\mathbf{x}}, h) + \Phi'(\bar{\mathbf{x}}, h)(\mathbf{x}_n - \bar{\mathbf{x}}) , \quad (6.2)$$

coincides with (6.1). Hence

$$e^{h_n F'(\bar{\mathbf{x}})} = \Phi'(\bar{\mathbf{x}}, h) , \quad \Phi(\bar{\mathbf{x}}, h) = \bar{\mathbf{x}} , \quad (6.3)$$

which means that  $\bar{\mathbf{x}}$  is a fixed point of the system  $\mathbf{x}_{n+1} = \Phi(\mathbf{x}_n, h)$ .  $\square$

Stability of numerical integrators can be roughly defined as follows: “the numerical solution provided by a stable numerical integrator does not tend to infinity when the exact solution is bounded” (see [3], p. 358). The integrator which is stable when applied to linear equations is said to be linearly stable. We may use the notion of A-stability (see, e.g., [18]): an integrator is said to be A-stable, if discretizations of all stable linear equations are stable as well.

Making one more assumption (quite natural, in fact) that the discretization of a linear system is linear, we see that locally exact integrators are linearly stable. Indeed, solutions of any locally exact discretization have the same trajectories as corresponding exact solutions.

**Corollary 6.2.** *Any locally exact numerical scheme is linearly stable and, in particular, A-stable.*

What is more, a locally exact discretization yields the best (exact) simulation of a linear equation in the neighbourhood of a fixed point. In particular, locally exact discretizations preserve any qualitative features of trajectories of linear equations (up to round-off errors, of course).



Numerical experiments show that locally exact schemes are exceptionally stable also for some simple nonlinear systems [8, 9, 10], but we have no theoretical results concerning the stability (e.g., algebraic stability [18]) in the nonlinear case.

In order to illustrate these general results we will apply four schemes presented above to a linear equation  $\dot{\mathbf{x}} = A\mathbf{x}$ . Locally exact explicit Euler scheme and locally exact implicit Euler scheme yield, respectively,

$$\mathbf{x}_{n+1} - \mathbf{x}_n = (e^{h_n A} - I) \mathbf{x}_n , \quad (6.4)$$

$$\mathbf{x}_{n+1} - \mathbf{x}_n = (I - e^{-h_n A} - I) \mathbf{x}_{n+1} . \quad (6.5)$$

Implicit midpoint and trapezoidal rules yield an identical equation, namely

$$\mathbf{x}_{n+1} - \mathbf{x}_n = \left( \tanh \frac{h_n A}{2} \right) (\mathbf{x}_{n+1} + \mathbf{x}_n) . \quad (6.6)$$

Simple calculations show that all resulting equations reduce to the exact discretization:  $\mathbf{x}_{n+1} = e^{h_n A} \mathbf{x}_n$ . In particular, if real part of any eigenvalue of  $A$  is negative, then  $\mathbf{x}_n \rightarrow 0$  for  $n \rightarrow \infty$ . A-stability is evident.

## 7 Numerical experiments

In order to illustrate performance of locally exact numerical schemes we consider circular orbits for the Hamiltonian

$$H(\mathbf{x}, \mathbf{p}) = \frac{1}{2} |\mathbf{p}|^2 + \frac{1}{2} |\mathbf{x}|^2 - \frac{1}{30} |\mathbf{x}|^3 , \quad (7.1)$$

where the exact solution can be easily found. Namely, initial conditions  $\mathbf{x}_0 = (x_0, 0)$ ,  $\mathbf{p}_0 = (0, x_0 \sqrt{1 - 0.1x_0})$  yield a circular orbit of radius  $R = x_0$ . Circular orbits exist for  $R < 10$ . We made numerical tests for  $R = 0.2$ ,  $R = 1$  and  $R = 5$  (corresponding periods of exact solutions are given by: 6.347, 6.623 and 8.886, respectively). While solving implicit algebraic equation for  $\mathbf{x}_{n+1}$  we used the fixed point method. Iterations were made until the accuracy  $3 \cdot 10^{-16}$  was reached. The maximal number of iterations was limited to 20. All figures present global error at  $t = 12.5$ .

Locally exact modifications improve the accuracy but are quite expensive. In our numerical experiments we use different time steps for different schemes

to assure the same computational cost (i.e., at all figures the computational cost in every column is the same). The numerical cost is estimated as a number of function evaluations. In other words, we use the scaled time step  $\tilde{h}$  such that  $h = \lambda \tilde{h}$ , where  $\lambda$  depend on numerical scheme and (to some extent) also on  $h$ . In order to evaluate the parameter  $\lambda$  we start from  $\lambda = 1$ , estimate the computational costs of considered schemes and multiply  $h$  by the relative computational cost. The procedure is repeated until computational costs become equal with accuracy about 1%. Approximate values of the parameter  $\lambda$  are given in figure captions.

In the case of Euler schemes (Figs. 1 and 2) the advantage of locally exact modifications is obvious. Although the cost of locally exact modifications is considerably higher, they perform much better than both standard Euler schemes. Actually the accuracy of all 3 modifications (taken with the same time step) is similar. However, taking into account the computational cost, we see that the scheme EEU-LEX is the best modification.

Locally exact modifications of implicit midpoint and trapezoidal rules increase have higher accuracy only for orbits with a relatively small radius (Figs. 3 and 4). The best results are produced by scheme IMP-LEX. For larger orbits the unmodified implicit midpoint rule is most accurate. Similar situation has place for gradient schemes. Locally exact modifications give a considerable improvement, by about 1-2 orders of magnitude, for trajectories in a quite large neighborhood of the stable equilibrium, see Figs. 5 and 6. For large  $R$  (e.g.,  $R = 5$ ) locally exact modifications improve considerably only GR-IA. Their influence on the GR-SYM is neutral or even negative. In fact GR-SYM has similar accuracy for all considered orbits while its locally exact modifications are very accurate only for smaller values of  $R$ .

## 8 Concluding remarks

We presented a new class of numerical schemes characterized by the so called local exactness. This notion is known (under different names) since almost fifty years. The original application, see [27], has been confined to the exact discretization of linearized equations, compare Section 2.3. We obtain in this way a particular locally exact integrator which has some advantages (e.g., it can serve as a good predictor, see [9], Section V.A).

Our approach has two new features. First, we modify known numerical schemes in a locally exact way. Any numerical scheme admits at least one

(usually more) natural locally exact modification, see Section 3. Second, we try to preserve geometric properties of the original numerical scheme. This task is not trivial. In this paper we present one successful application: locally exact modifications of discrete gradient methods for canonical Hamilton equations. We are able to exactly preserve the energy integral and, in the same time, increase the accuracy by many orders of magnitude. Another advantage is a variable time step. Unlike symplectic methods (which work mostly for the constant time step) discrete gradient methods admit conservative modifications with variable time step. Therefore, one may easily implement any variable step method in order to obtain further improvement.

In the one-dimensional case proposed modifications, although a little bit more expensive, turns out to be more accurate even by 8 orders of magnitude in comparison to the standard discrete gradient scheme, see [9, 10]. In multidimensional cases the relative cost of our algorithm is higher, but still our method is of great advantage for orbits in the neighborhood of the stable equilibrium. The accuracy is increased by 1-2 orders of magnitude. We point out that in most cases locally exact modifications do not change orders of modified schemes.

We presented locally exact modifications from a unified theoretical perspective. There are many possible further developments. First of all, we plan to apply our approach to chosen multidimensional problems, testing the accuracy of locally exact schemes by numerical experiments. Then, we would like to extend the range of applications. It would be natural to consider locally exact modifications of Runge-Kutta schemes and  $G$ -symplectic integrators [2]. However, our first attempts seem to suggest that this is a challenging problem. Modifications proposed in this paper contain exponentials of variable matrices. Similar time-consuming evaluations are characteristic for all exponential integrators and in this context effective methods of computing matrix exponentials have been recently developed [15, 26].

Throughout this paper we assumed the autonomous case,  $\dot{\mathbf{x}} = F(\mathbf{x})$ . The extension on the non-autonomous case can be done along lines indicated already in Pope's paper [27]. A separate problem is to obtain in this case any locally exact modification with geometric properties. Another open problem is the construction of locally exact (or, at least, linearization preserving) deformations of generalized discrete gradient algorithms preserving all first integrals (see [22]). Linearization-preserving integrators form an important subclass of locally exact schemes. It would be worthwhile to study linearization-preserving modifications of geometric numerical integrators, es-

pecially in those cases when locally exact are difficult or impossible to construct.

**Acknowledgment.** Research supported in part by the National Science Centre (NCN) grant no. 2011/01/B/ST1/05137.

## References

- [1] R.P.Agarwal: *Difference equations and inequalities* (Chapter 3), Marcel Dekker, New York 2000.
- [2] J.C.Butcher: “Numerical methods for ordinary differential equations”, second edition, Wiley & Sons, Chichester 2008.
- [3] E.Celledoni, D.Cohen, B.Owren: “Symmetric exponential integrators with an application to the cubic Schrödinger equation”, *Found. Comput. Math.* **8** (2008) 303-317.
- [4] J.L.Cieśliński: “An orbit-preserving discretization of the classical Kepler problem”, *Phys. Lett. A* **370** (2007) 8-12.
- [5] J.L.Cieśliński: “Comment on ‘Conservative discretizations of the Kepler motion’ ”, *J. Phys. A: Math. Theor.* **43** (2010) 228001 (4pp).
- [6] J.L.Cieśliński: “On the exact discretization of the classical harmonic oscillator equation”, *J. Difference Equ. Appl.* **17** (2011) 1673-1694.
- [7] J.L.Cieśliński, B.Ratkiewicz: “On simulations of the classical harmonic oscillator equation by difference equations”, *Adv. Difference Eqs.* **2006** (2006) 40171 (17pp).
- [8] J.L.Cieśliński, B.Ratkiewicz: “Long-time behaviour of discretizations of the simple pendulum equation”, *J. Phys. A: Math. Theor.* **42** (2009) 105204 (29pp).
- [9] J.L.Cieśliński, B.Ratkiewicz: “Improving the accuracy of the discrete gradient method in the one-dimensional case”, *Phys. Rev. E* **81** (2010) 016704 (6pp).
- [10] J.L.Cieśliński, B.Ratkiewicz: “Energy-preserving numerical schemes of high accuracy for one-dimensional Hamiltonian systems”, *J. Phys. A: Math. Theor.* **44** (2011) 155206 (14pp).
- [11] J.L.Cieśliński, B.Ratkiewicz: “Discrete gradient algorithms of high-order for one-dimensional systems”, *Comp. Phys. Comm.* **183** (2012) 617-627.
- [12] O.Gonzales: “Time integration and discrete Hamiltonian systems”, *J. Nonl. Sci.* **6** (1996) 449-467.
- [13] D.Greenspan: “An algebraic, energy conserving formulation of classical molecular and Newtonian  $n$ -body interaction”, *Bull. Amer. Math. Soc.* **79** (1973) 432-427.
- [14] E.Hairer, C.Lubich, G.Wanner: *Geometric numerical integration: structure-preserving algorithms for ordinary differential equations*, second edition, Springer, Berlin 2006.
- [15] M.Hochbruck, Ch.Lubich: “On Krylov subspace approximations to the matrix exponential operator”, *SIAM J. Numer. Anal.* **34** (1997) 1911-1925.
- [16] M.Hochbruck, Ch.Lubich: “A Gautschi-type method for oscillatory second-order differential equations”, *Numer. Math.* **83** (1999) 403-426.

- [17] A.Iserles: “Insight, not just numbers”, *Proceedings of the 15th IMACS World Congress*, vol. II, ed. by A.Sydow, pp. 589-594; Wissenschaft & Technik Verlag, Berlin 1997.
- [18] A.Iserles: *A first course in the numerical analysis of differential equations*, second edition, Cambridge Univ. Press 2009.
- [19] T.Itoh, K.Abe: “Hamiltonian conserving discrete canonical equations based on variational difference quotients”, *J. Comput. Phys.* **77** (1988) 85-102.
- [20] R.A.LaBudde, D.Greenspan: “Discrete mechanics – a general treatment”, *J. Comput. Phys.* **15** (1974) 134-167.
- [21] D.J.Lawson: “Generalized Ruge-Kutta processes for stable systems with large Lipschitz constants”, *SIAM J. Numer. Anal.* **4** (1967) 372-380.
- [22] R.I.McLachlan, G.R.W.Quispel, N.Robidoux: “Geometric integration using discrete gradients”, *Phil. Trans. R. Soc. London A* **357** (1999) 1021-1045.
- [23] R.I.McLachlan, G.R.W.Quispel, P.S.P.Tse: “Linearization-preserving self-adjoint and symplectic integrators”, *BIT Numer. Math.* **49** (2009) 177-197.
- [24] R.E.Mickens: *Nonstandard finite difference models of differential equations*, World Scientific, Singapore 1994.
- [25] B.V.Minchev, W.M.Wright: “A review of exponential integrators for first order semi-linear problems”, *preprint NTNU/Numerics/N2/2005*, Trondheim 2005.
- [26] J.Niesen, W.M.Wright: “Algorithm 919: a Krylov subspace algorithm for evaluating the  $\varphi$ -functions appearing in exponential integrators”, *ACM Trans. Math. Software* **38** (3) (2012), article 22.
- [27] D.A.Pope: “An exponential method of numerical integration of ordinary differential equations”, *Commun. ACM* **6** (8) (1963) 491-493.
- [28] R.B.Potts: “Differential and difference equations”, *Am. Math. Monthly* **89** (1982) 402-407.
- [29] G.R.W.Quispel, G.S.Turner: “Discrete gradient methods for solving ODE’s numerically while preserving a first integral”, *J. Phys. A: Math. Gen.* **29** (1996) L341-L349.

Figure 1: Error of numerical solutions at  $t = 12.5$  for a circular orbit ( $R = 0.2$ ) as a function of scaled time step  $\tilde{h}$  (i.e.,  $h = \lambda\tilde{h}$ ). EEU: white discs ( $\lambda = 1$ ), IEU: white squares ( $\lambda = 24, 30, 42, 60$ , respectively), EEU-LEX: gray discs ( $\lambda = 8$ ), IEU-LEX: gray squares ( $\lambda = 34, 40, 53, 67$ ), IEU-ILEX: black squares ( $\lambda = 130, 200, 200, 200$ ).

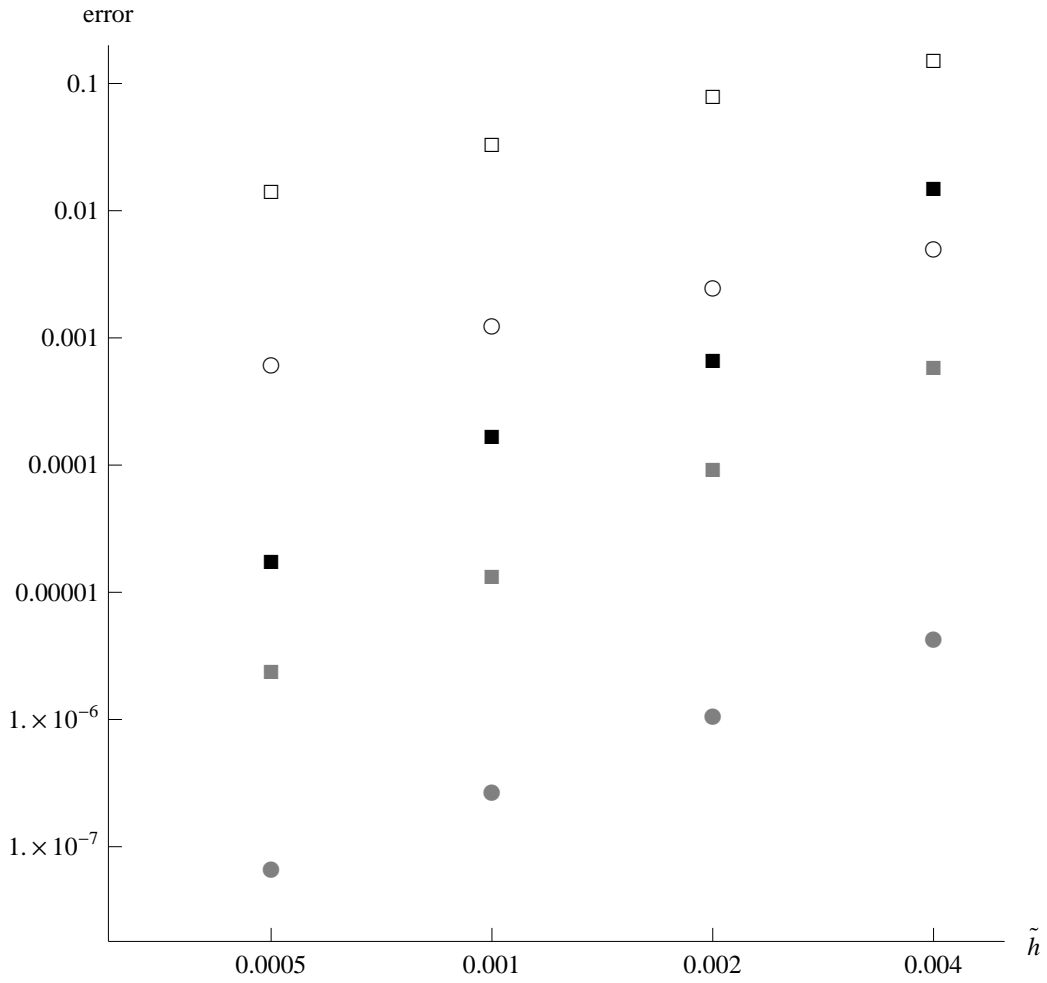


Figure 2: Error of numerical solutions at  $t = 12.5$  for a circular orbit ( $R = 5$ ) as a function of scaled time step  $\tilde{h}$  (i.e.,  $h = \lambda\tilde{h}$ ). EEU: white discs ( $\lambda = 1$ ), IEU: white squares ( $\lambda = 25, 31, 42, 60$ ), EEU-LEX: gray discs ( $\lambda = 8$ ), IEU-LEX: gray squares ( $\lambda = 34, 40, 52, 67$ ), IEU-ILEX: black squares ( $\lambda = 200$ ).

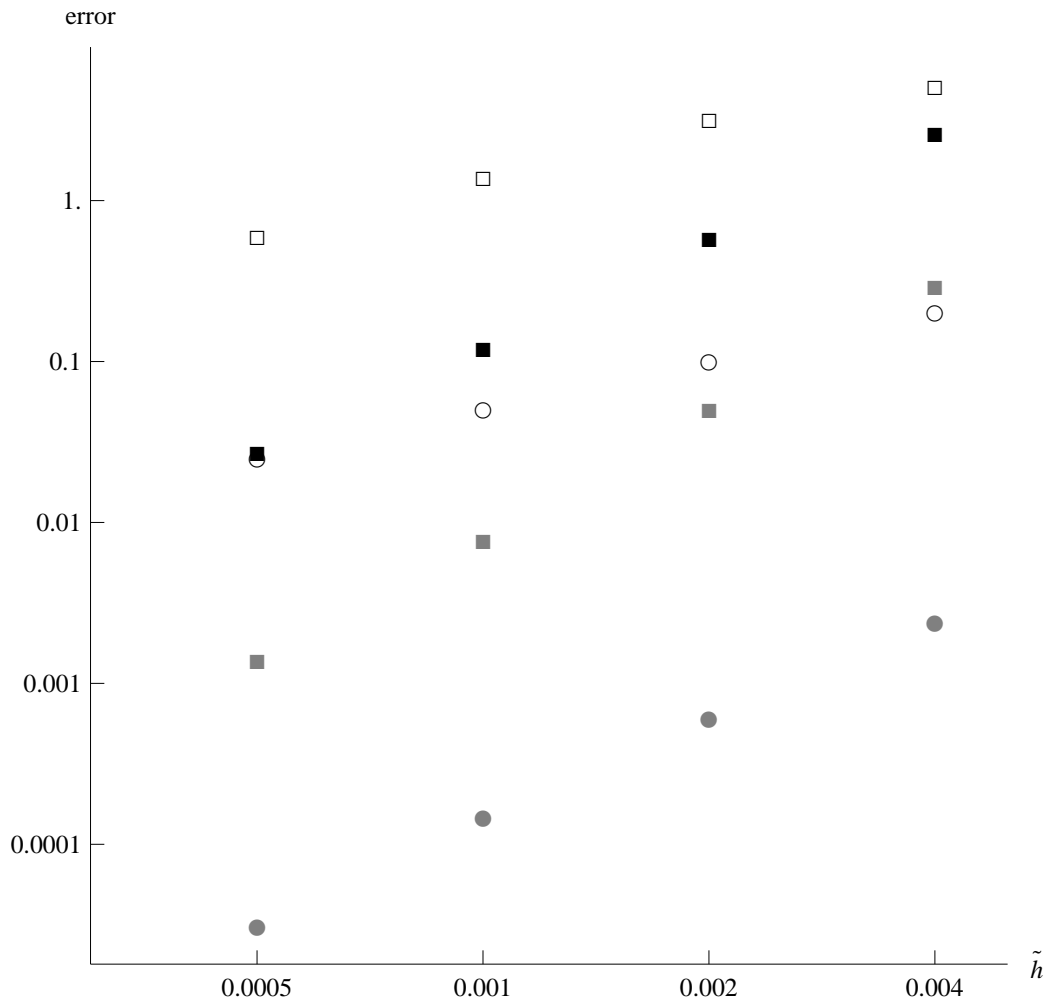


Figure 3: Error of numerical solutions at  $t = 12.5$  for a circular orbit ( $R = 0.2$ ) as a function of scaled time step  $\tilde{h}$  (i.e.,  $h = \lambda\tilde{h}$ ). IMP: white discs ( $\lambda = 1$ ), IMP-LEX: gray discs ( $\lambda = 1.3, 1.3, 1.3, 1.2$ ), IMP-SLEX: black discs ( $\lambda = 3.4, 3.4, 3.9, 3.9$ ), TR: white squares ( $\lambda = 1.06$ ), TR-LEX: gray squares ( $\lambda = 1.9, 1.6, 1.8, 1.8$ ), TR-SLEX: black squares ( $\lambda = 3.5, 3.4, 4.0, 3.9$ ).

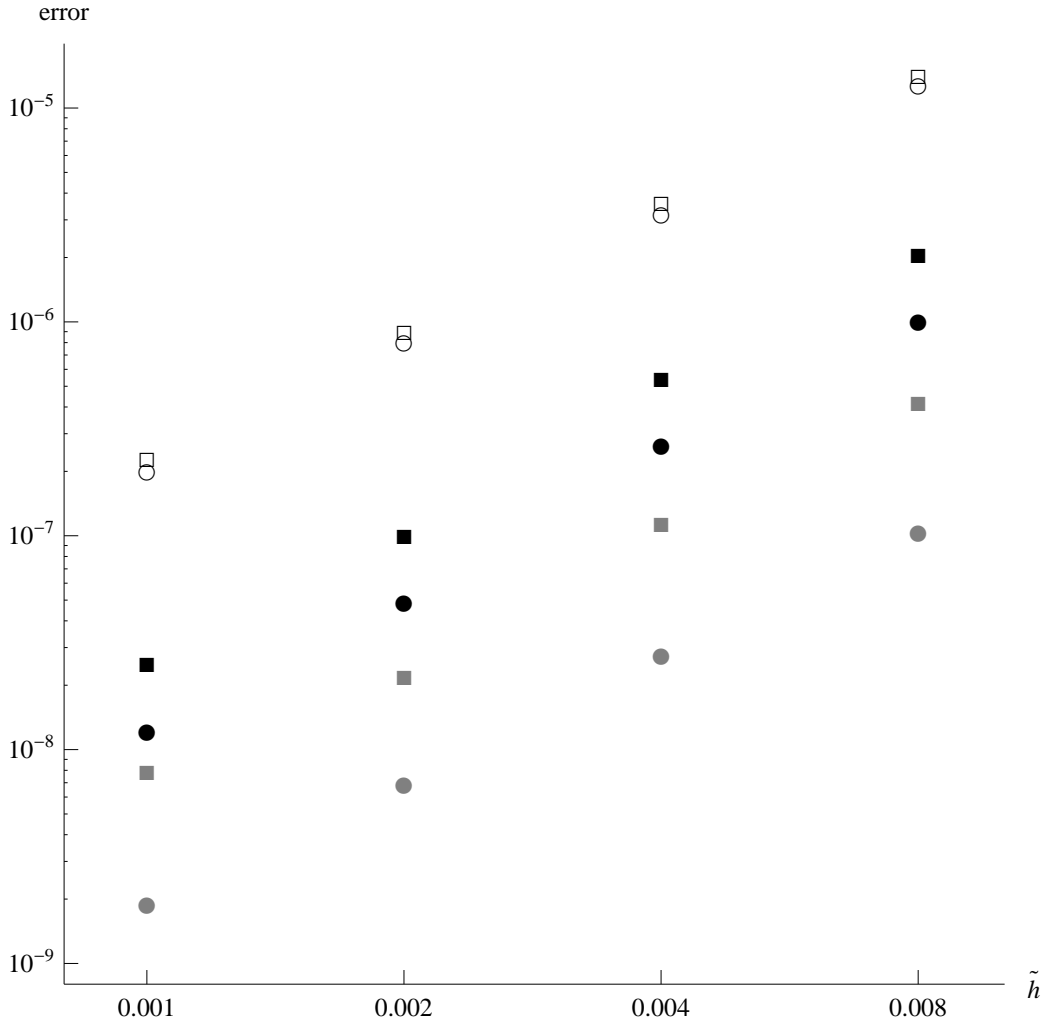




Figure 4: Error of numerical solutions at  $t = 12.5$  for a circular orbit ( $R = 1$ ) as a function of scaled time step  $\tilde{h}$  (i.e.,  $h = \lambda\tilde{h}$ ). IMP: white discs ( $\lambda = 1$ ), IMP-LEX: gray discs ( $\lambda = 1.5, 1.3, 1.4, 1.2$ ), IMP-SLEX: black discs ( $\lambda = 3.4, 3.4, 3.9, 3.9$ ), TR: white squares ( $\lambda = 1.06$ ), TR-LEX: gray squares ( $\lambda = 1.9, 1.6, 1.8, 1.8$ ), TR-SLEX: black squares ( $\lambda = 3.5, 3.4, 4.0, 3.9$ ).

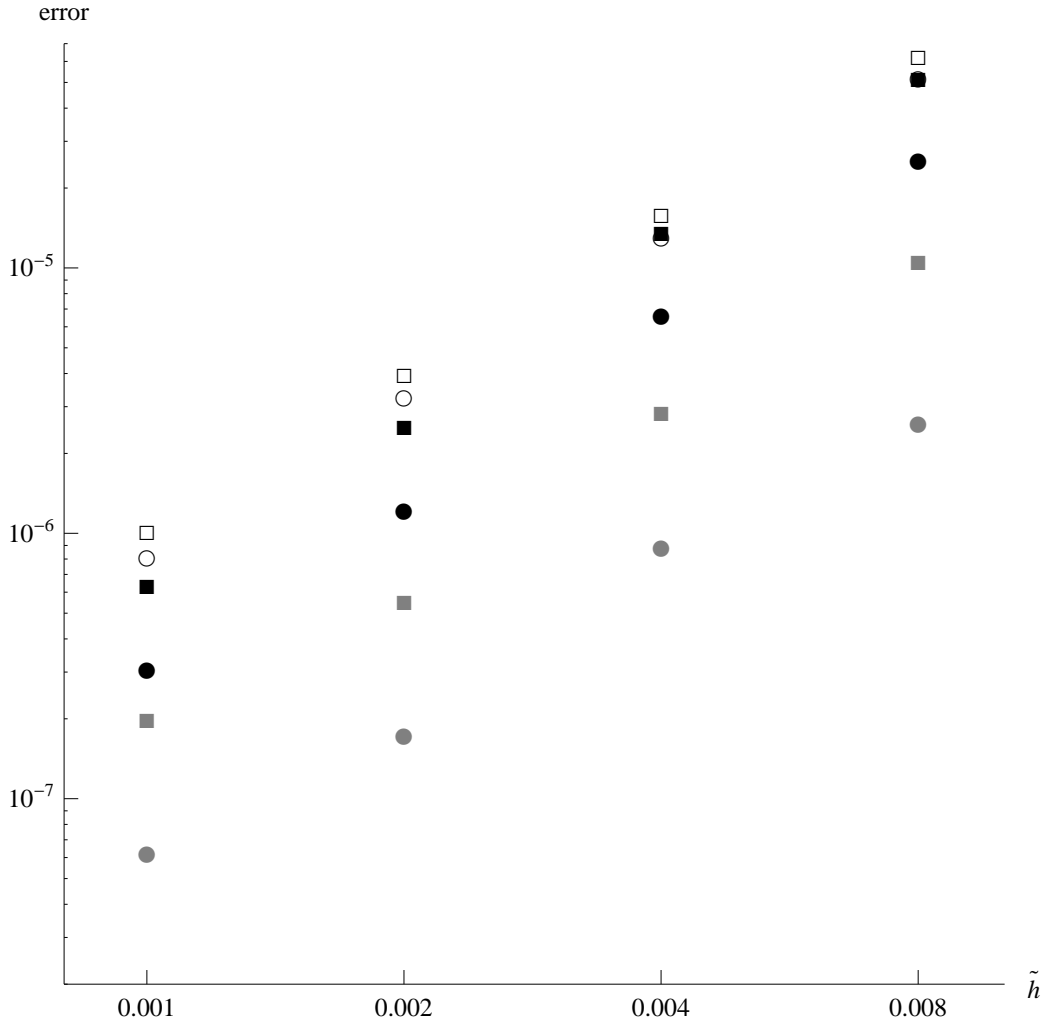


Figure 5: Error of numerical solutions at  $t = 12.5$  for a circular orbit ( $R = 0.2$ ) as a function of scaled time step  $\tilde{h}$  (i.e.,  $h = \lambda\tilde{h}$ ). GR-IA: white discs, GR-IA-LEX: gray discs, GR-IA-SLEX: black discs, GR-SYM: white squares, GR-SYM-LEX: gray squares, GR-SYM-SLEX: black squares.

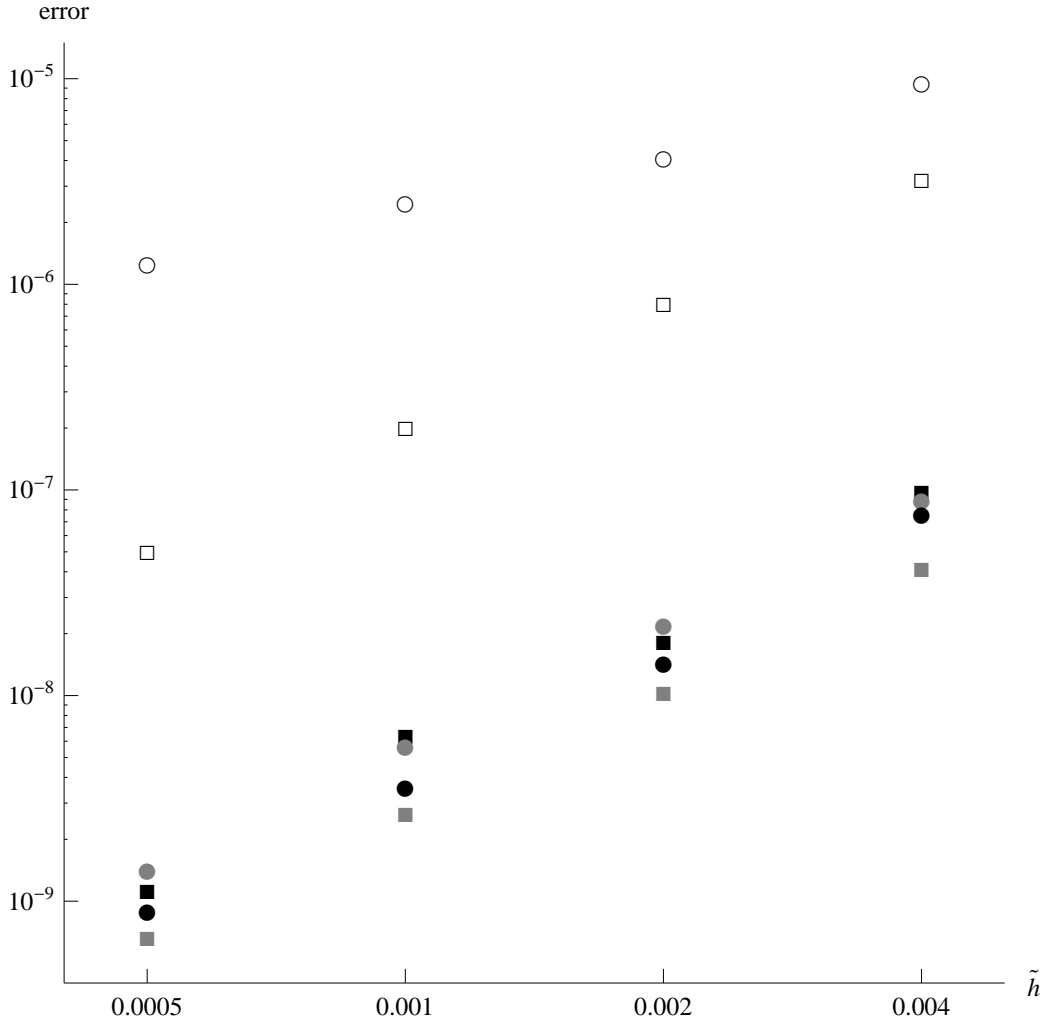


Figure 6: Error of numerical solutions at  $t = 12.5$  for a circular orbit ( $R = 1$ ) as a function of scaled time step  $\tilde{h}$  (i.e.,  $h = \lambda\tilde{h}$ ). GR-IA: white discs ( $\lambda = 0.8$ ), GR-IA-LEX: gray discs ( $\lambda = 1.1$ ), GR-IA-SLEX: black discs ( $\lambda = 1.5, 1.8, 1.5, 1.7$ ), GR-SYM: white squares ( $\lambda = 1$ ), GR-SYM-LEX: gray squares ( $\lambda = 1.3, 1.4, 1.3, 1.4$ ), GR-SYM-SLEX: black squares ( $\lambda = 1.7, 2.0, 1.7, 1.9$ ).

